

IN THE CLAIMS

Claims 1 (canceled)

Claim 2 (previously presented): A method of launching a software application in a hand-held device, comprising:

- receiving an abbreviated textual command in a natural language search engine;
- while receiving the abbreviated textual command, performing the steps of:
 - searching a natural language database that stores a data set of abbreviated textual commands and associated application commands;
 - analyzing historical preferences to determine one or more probable complete commands matching a currently received portion of the abbreviated textual command; and
 - displaying a list of probable complete commands matching the currently received portion of the abbreviated textual command.

Claim 3 (previously presented): The method of claim 2, comprising the additional step of:

- if a user selects a complete command from the list, then setting the complete command as the abbreviated textual command, and executing the associated application command.

Claim 4 (previously presented): The method of claim 2, comprising the additional step of:

- if a user does not select a complete command from the list, then receiving an entire abbreviated textual command in the natural language search engine.

Claim 5 (previously presented): The method of claim 4, further comprising:

- if the abbreviated textual command has an exact match in the data set, then setting the exact match as a user command;

- if the abbreviated textual command does not have an exact match in the data set, then analyzing historical preferences to determine if the abbreviated textual command has a probable match in the data set;

- if the abbreviated textual command has a probable match in the data set, then setting the probable match as the user command;

if the abbreviated textual command does not have a probable match in the data set, then presenting a list of possible commands, receiving a command choice, and setting the command choice as the user command; and
executing the user command.

Claim 6 (previously presented): The method of claim 2, wherein the step of analyzing historical preferences is performed using a set of probability factors that are generated based on historical preferences, where the abbreviated textual command has a probable match in the data set when a probability factor associated with the probable match is greater than a predetermined value.

Claim 7 (previously presented): The method of claim 6, wherein the predetermined value is defined by a user.

Claim 8 (previously presented): The method of claim 6, comprising the additional step of:
adjusting the set of probability factors each time the abbreviated textual command is entered into the hand-held device.

Claim 9 (previously presented): The method of claim 2, wherein:
the abbreviated textual command has a first component and a second component, wherein the first component represents a desired application command, and the second component represents a desired application tag; and
the natural language database stores a data set of abbreviated textual commands and associated application commands and tags.

Claim 10 (previously presented): The method of claim 2, wherein the abbreviated textual command is entered into a graphical dialog box.

Claim 11 (previously presented): The method of claim 2, wherein the natural language search engine can receive the abbreviated textual command while any of the software applications are executing.

Claim 12 (previously presented): The method of claim 5, wherein the list of possible commands presented if the abbreviated textual command does not have a probable match in the data set includes a set of recently executed application commands.

Claim 13 (previously presented): The method of claim 5, wherein the list of possible commands presented if the abbreviated textual command does not have a probable match in the data set includes a set of generic application commands that the natural language search engine is capable of executing.

Claim 14 (previously presented): A hand-held device, comprising:

- a plurality of software applications;
- an input device;
- a natural language search engine operable to receive a two-part keystroke combination from the input device, the two-part keystroke combination having a first component and a second component;
- the natural language search engine being further operable to match the first component with a desired application command, match the second component with a desired application tag, execute the desired application command, and retrieve data associated with the application command using the desired application tag.

Claim 15 (previously presented): The hand-held device of claim 14, wherein executing the application command launches a software application.

Claim 16 (previously presented): The hand-held device of claim 14, further comprising:

- a natural language database configured to store a data set of keystroke combinations and associated application commands, the natural language database being used by the natural language search engine to match the keystroke combination with the desired application command.

Claim 17 (previously presented): The hand-held device of claim 14, wherein the data set includes probability factors that represent a probability that the application command is desired by a user when a corresponding keystroke combination is entered.

Claim 18 (previously presented): The hand-held device of claim 14, wherein the natural language search engine can receive keystroke combinations while any of the software applications are executing.

Claim 19 (previously presented): The hand-held device of claim 14, wherein the keystroke combination is entered into a graphical dialog box.

Claim 20 (previously presented): The hand-held device of claim 14, further comprising:
a home screen that is a graphical interface between a user and the natural language search engine.

Claim 21 (previously presented): The hand-held device of claim 20, wherein the home screen includes an icon ribbon having a plurality of icons, and wherein a user may launch one of the software applications by either selecting one of the icons or entering a keystroke combination.

Claim 22 (previously presented): The hand-held device of claim 14, wherein the natural language search engine is operative to match the keystroke combination with the desired application command by presenting a user with a list of likely command choices.

Claim 23. (previously presented) The hand-held device of claim 14, wherein the input device is a trackwheel.

Claims 24-27 (canceled)

Claim 28 (previously presented): In a mobile device having a graphical input device and a textual input device, a method comprising:

- displaying an icon ribbon having a plurality of icons on the graphical input device;
- if a user selects one of the icons via the graphical input device, then executing an application command associated with the icon; and
- if the user enters a two-part abbreviated textual command via the textual input device, then receiving the two-part abbreviated textual command in a natural language search engine, matching the first component with a desired application command, matching the second component with a

desired application tag, executing the desired application command, and retrieving data associated with the application command using the desired application tag.

Claim 29 (previously presented): The method of claim 28, wherein a natural language database stores a data set of abbreviated textual commands and associated application commands and tags.

Claim 30 (previously presented): The method of claim 28, wherein the abbreviated textual command is entered into a graphical dialog box.

Claim 31 (previously presented): The method of claim 28, wherein the natural language search engine can receive the abbreviated textual command while any of the software applications are executing.

Claim 32 (previously presented): The method of claim 28, wherein the graphical input device is a thumbwheel.

Claims 33-36 (canceled)

Claim 37 (new): A method comprising:

- storing a data set of abbreviated textual commands and corresponding complete commands;
- receiving a portion of an abbreviated textual command; and
- before receiving the entire abbreviated textual command, comparing the received portion of the abbreviated textual command to the stored abbreviated commands to determine a probable subset of the complete commands.

Claim 38 (new): The method of claim 37 further comprising after the comparing step:

- displaying the probable subset of the complete commands to the user.

Claim 39 (new): The method of claim 38 further comprising after the displaying step:

- receiving an indication of which of the displayed complete commands a user chooses; and
- executing the chosen complete command.

Claim 40 (new): The method of claim 38 further comprising after the displaying step:
receiving a further portion of the abbreviated textual command; and
narrowing the probable subset based on the further portion received.

Claim 41 (new): The method of claim 37 further comprising:
when the probable subset consists of only one complete command, executing that one
complete command.

Claim 42 (new): The method of claim 37 wherein the storing step includes a user assigning which
complete commands should correspond in the future to which abbreviated textual commands.

Claim 43 (new): The method of claim 37 wherein the storing step includes generating the data set
based on which abbreviated textual commands a user has historically used for choosing each
complete command.

Claim 44 (new): The method of claim 37 wherein the comparing step includes:
if the data set indicates that the user has chosen to execute a particular complete command
more than a predetermined percentage of the time less than 100% after having entered an
abbreviated textual command matching the currently received portion of text, then narrowing the
subset to that command.

Claim 45 (new): The method of claim 44 wherein the predetermined percentage is 50%.

Claim 46 (new): A method comprising:
receiving a textual command that includes a first text component representing an operation
and a second text component representing an object, the second component being entered after the
first component and separated from the first component by a delimiter; and
performing the operation with the object;
the receiving and performing being performed by an electronic device.

Claim 47 (new): The method of claim 46 wherein the delimiter is a space.